# Requirement guide for class project

[Use cases](https://en.wikipedia.org/wiki/Use_case) will be used as requirements for this project.

## Template

An adaptation of the standard Cockburn template will be used. The template and examples follow:

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-1:Vizualizare produse | | |
| Primary actor | Agent de vanzari | Secondary actors |  |
| Description | Agentul de vanzare deschide aplicatia de produse. Din meniul afisat el alege vizualizare produse apoi pe ecran ii va aparea o lista cu toate produsele si cu stocul fiecaruia. | | |
| Trigger | Un agent de vanzari doreste sa vada o lista de produse. | | |
| Preconditions | Agentul de vanzari este autentificat in aplicatie. | | |
| Postconditions | - | | |
| Normal flow | Vizualizare toate produse   1. Agentul de vanzari vede meniul aplicatiei 2. Din meniul aplicatiei alege butonul cu "Vizualizare produse" 3. O lista cu toate produsele apare si el poate sa vada stocul la fiecare | | |
| Alternative flows | - | | |
| Exceptions | 1. Serverul este cazut si nu poate afisa toate produsele | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-2:Vizualizare istoric comenzi | | |
| Primary actor | Agent de vanzari | Secondary actors |  |
| Description | Agentul de vanzare deschide aplicatia de produse. Din meniul afisat el alege Istoric comenzi apoi pe ecran ii va aparea o lista cu toate comenzile efectuate de catre agentul respectiv. | | |
| Trigger | Un agent de vanzari doreste sa vada istoricul lui de comenzi. | | |
| Preconditions | PRE-1. Agentul de vanzari este autentificat in aplicatie. | | |
| Postconditions | - | | |
| Normal flow | Vizualizare istoric comenzi   1. Agentul de vanzari vede meniul aplicatiei 2. Din meniul aplicatiei alege butonul cu "Istoric comenzi" 3. O lista cu toate comenzile efectuate de catre el apare. | | |
| Alternative flows | Cand agentul nu a comandat nimic pana in prezent  Vizualizare istoric comenzi   1. Agentul de vanzari vede meniul aplicatiei 2. Din meniul aplicatiei alege butonul cu "Istoric comenzi" 3. Apare mesajul "Nu aveti comenzi efectuate in trecut" | | |
| Exceptions | EC-1.Serverul este cazut si nu poate afisa istoricul comenzilor. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-3:Comanda un produs | | |
| **Primary actor** | Agent de vanzari | **Secondary actors** |  |
| **Description** | Agentul de vanzari vede lista cu toate produsele. In dreptul produsului pe care vrea sa il comande apasa pe butonul comanda. O fereastra se deschide si ii va cere informatii de livrare si de facturare. Agentul poate sa introduca si o cantitate mai mare daca doreste. | | |
| **Trigger** | Un agent de vanzari doreste sa comande un produs din aplicatie | | |
| **Preconditions** | PRE-1. Agentul de vanzari este autentificat in aplicatie.  PRE-2. Agentul de vanzari a apasat pe butonul "Vizualizare produse" si se vede lista cu toate produsele disponibile si cantitatile lor. | | |
| **Postconditions** | POST-1. Comanda este salvata in sistem cu statusul "In curs de procesare"  POST-2. Lista tuturor produselor este actualizata cu noua cantitate din produsul care a fost comandat. | | |
| **Normal flow** | **1.0 Comanda un singur produs.**   1. Agentul apasa pe butonul de "Comanda produs" 2. Se deschide o pagina noua in care agentul trebuie sa introduca detaliile de livrare si de facturare.De asemenea introduce si datele de plata. 3. Agentul introduce datele de livrare 4. Agentul introduce datele de facturare 5. Agentul poate introduce o cantitate din produsului curent. 6. Valoarea totala a comenzii este calculata si afisata pe ecran. 7. Agentul introduce informatiile de plata a comenzii. 8. Agentul confirma comanda. 9. O noua pagina se deschide unde agentul poate vedea confirmarea comenzii si factura. | | |
| **Alternative flows** | **1.1 Comanda mai multe unitati din acelasi produs.**   1. Agentul alege o cantitate mai mare decat 1 a produsului 2. Intoarce-te la pasul 6 din normal flow.   **1.2 Comanda un alt produs**   1. Agentul se intoarce in pagina cu lista de produse disponibile. 2. Intoarce-te la pasul 1 din normal flow. | | |
| **Exceptions** | **1.0.E1 Stocul este limitat iar cineva face o comanda mai rapida din ultima unitate din produsul respectiv**  1. Apare pe ecran mesajul cu "Stoc epuizat de altcineva".  **1.0.E2 Metoda de plata nu este acceptata de sistem.**  1. Agentul introduce o metoda de plata invalida(ex AmericanExpress) care este automat refuzata chiar daca agentul are fonduri.  2a. Agentul este rugat sa inchida fereastra si sa incerce sa comande din nou produsul folosind o alta metoda de plata. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-4: Autentificare | | |
| **Primary actor** | Agent de vanzari | **Secondary actors** | Manager |
| **Description** | Un agent de vanzari deschide aplicatia si doreste sa se autentifice.Aplicatia nu se poate accesa fara un cont activ.Agentul de vanzari isi introduce username-ul si parola apoi apasa pe butonul de login. Acelasi scenariu si pentru un manager. | | |
| **Trigger** | Un agent de vanzari doreste sa se autentifice in aplicatie. | | |
| **Preconditions** | None | | |
| **Postconditions** | None | | |
| **Normal flow** | **1.0 Autentificare utilizator**   1. Agentul de vanzari deschide aplicatia. 2. Agentul de vanzari introduce username si parola. 3. Aplicatia verifica daca datele introduse exista in baza de date 4. Se deschide pagina principala corespunzatoare tipului de utilizator al aplicatiei. | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Datele de autentificare nu sunt introduse corect atunci apare un mesaj de eroare.**  **EC-1: Server-ul nu este functional si nu se poate face conectarea.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-5: Returnare/Anulare comanda | | |
| **Primary actor** | Agent de vanzari | **Secondary actors** |  |
| **Description** | Un agent de vanzari doreste sa anuleze sau sa returneze o comanda facuta.El merge in istoricul comenzilor si in dreptul fiecarei comenzi apare un buton de returnare/anulare comanda. | | |
| **Trigger** | Un agent de vanzari doreste sa returneze/anuleze o comanda. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie  PRE-2: Agentul de vanzari se afla pe pagina cu istoric comenzi. | | |
| **Postconditions** | POST-1.0: Daca comanda inca este in procesare si nu a fost confirmata de catre un manager atunci dupa ce se anuleaza comanda stocul produsului va reveni inapoi la cantintatea de dinainte de comanda.  POST-1.1: Daca comanda a fost confirmata de catre un manager atunci stocul nu va fi actualizat imediat.Un manager trebuie sa confirme returul. | | |
| **Normal flow** | **1.0 Anulare/Returnare comanda**   1. Agentul de vanzari se afla pe pagina cu istoric comenzi. 2. Agentul de vanzari apasa pe butonul de anulare/returnare comanda. 3. Se deschide o noua fereastra unde se vad detaliile comenzii.La sfarsit de pagina agentul trebuie sa introduca motivul returnarii/anularii. 4. Agentul apasa pe confirmare anulare/returnare. 5. Apare un mesaj de confirmare | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Caz special: Daca se incearca anularea unei comenzi in timp ce ea este confirmata de catre un manager.Apare un mesaj de eroare.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-6: Returnare/Anulare comanda | | |
| **Primary actor** | Agent de vanzari | **Secondary actors** |  |
| **Description** | Un agent de vanzari doreste sa anuleze sau sa returneze o comanda facuta.El merge in istoricul comenzilor si in dreptul fiecarei comenzi apare un buton de returnare/anulare comanda. | | |
| **Trigger** | Un agent de vanzari doreste sa returneze/anuleze o comanda. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie  PRE-2: Agentul de vanzari se afla pe pagina cu istoric comenzi. | | |
| **Postconditions** | POST-1.0: Daca comanda inca este in procesare si nu a fost confirmata de catre un manager atunci dupa ce se anuleaza comanda stocul produsului va reveni inapoi la cantintatea de dinainte de comanda.  POST-1.1: Daca comanda a fost confirmata de catre un manager atunci stocul nu va fi actualizat imediat.Un manager trebuie sa confirme returul.  POST-2.0: In cazul anularii comenzii statusul comenzii va trece din “In procesare” in “Anulata”.  POST-2.1: In cazul returnarii unei comenzi statusul ei va trece din “Confirmata” in “Asteptare confirmare retur” | | |
| **Normal flow** | **1.0 Anulare/Returnare comanda**   1. Agentul de vanzari se afla pe pagina cu istoric comenzi. 2. Agentul de vanzari apasa pe butonul de anulare/returnare comanda. 3. Se deschide o noua fereastra unde se vad detaliile comenzii.La sfarsit de pagina agentul trebuie sa introduca motivul returnarii/anularii. 4. Agentul apasa pe confirmare anulare/returnare. 5. Apare un mesaj de confirmare | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Caz special: Daca se incearca anularea unei comenzi in timp ce ea este confirmata de catre un manager.Apare un mesaj de eroare.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-7: Vizualizare comenzi primite | | |
| **Primary actor** | Manager | **Secondary actors** |  |
| **Description** | Un manager de firma poate sa vada comenzile primite de la agentii de vanzari. | | |
| **Trigger** | Un manager doreste sa vada comenzile primite. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie | | |
| **Postconditions** | None | | |
| **Normal flow** | **1.0 Vizualizare comenzi primite**   1. Managerul apasa butonul de “Vizualizare comenzi primite” din meniu. 2. Se deschide o noua fereastra unde se gasesc toate comenzile primite fie ele deja confirmate fie comenzi care trebuie confirmate. | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Server-ul nu functioneaza iar nu se poate vedea lista comenzilor primite.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-8: Vizualizare returnari | | |
| **Primary actor** | Manager | **Secondary actors** |  |
| **Description** | Un manager poate sa vizualizeze cererile de returnare ale unor produse. | | |
| **Trigger** | Un manager doreste sa vada lista cu cererile de returnari. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie | | |
| **Postconditions** | None | | |
| **Normal flow** | **1.0 Vizualizare returnari**   1. Managerul apasa butonul de “Vizualizare returnari” din meniu. 2. Se deschide o fereastra unde apar toate cererile de returnare primite. | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Server-ul nu functioneaza si nu poate afisa lista** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-9: Confirmare comanda | | |
| **Primary actor** | Manager | **Secondary actors** |  |
| **Description** | Un manager poate sa confirme o comanda primita de la un agent de vanzari al unui produs | | |
| **Trigger** | Un manager doreste sa confirme o comanda primita de la un agent de vanzare. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie  PRE-2: Agentul de vanzari se afla pe pagina cu toate comenzile primite. | | |
| **Postconditions** | POST-1.0: Statusul comenzii va trece din stadiul “In procesare” la stadiul de “Confirmata” in baza de date. | | |
| **Normal flow** | **1.0 Confirmare comanda**   1. Managerul se afla pe pagina de comenzi primite. 2. Managerul vede in dreptul comenzilor care nu au fost confirmate un buton de confirmare. 3. Se deschide o noua fereastra unde vor fi mai multe detalii de confirmare sau de livrare(etc,). La sfarsit de pagina este butonul de confirmare 4. Managerul apasa pe confirmare. 5. Apare un mesaj de succes | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Caz special: Daca se incearca confirmarea unei comenzi in timp ce ea este anulata de catre un agent de vanzari.Apare un mesaj de eroare.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-10: Confirmare returnare comanda | | |
| **Primary actor** | Manager | **Secondary actors** |  |
| **Description** | Un manager poate sa confirme o cerere de returnare a unei comenzi confirmate anterior primita de la un agent de vanzari al unui produs. | | |
| **Trigger** | Un manager doreste sa confirme un retur al unei comenzi confirmate. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie  PRE-2: Agentul de vanzari se afla pe pagina cu toate cererile de returnare. | | |
| **Postconditions** | POST-1.0: Statusul comenzii va trece din stadiul “Confirmata” la stadiul de “Returnata” in baza de date. | | |
| **Normal flow** | **1.0 Confirmare returnare comanda**   1. Managerul se afla pe pagina de comenzi primite. 2. Managerul vede in dreptul comenzilor care nu au fost confirmate un buton de confirmare. 3. Se deschide o noua fereastra unde vor fi mai multe detalii de confirmare sau de livrare(etc,). La sfarsit de pagina este butonul de confirmare 4. Managerul apasa pe confirmare. 5. Apare un mesaj de succes | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Caz special: Daca se incearca confirmarea unei comenzi in timp ce ea este anulata de catre un agent de vanzari.Apare un mesaj de eroare.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | UC-11: Adaugare stoc al unui produs | | |
| **Primary actor** | Manager | **Secondary actors** |  |
| **Description** | Un manager poate sa adauge stoc pentru un produs existent in baza de date sau poate sa adauge un produs nou. | | |
| **Trigger** | Un manager doreste sa adauge stoc unui produs existent sau sa adauge un produs nou. | | |
| **Preconditions** | PRE-1: Agentul de vanzari este autentificat in aplicatie | | |
| **Postconditions** | POST-1.0: In cazul unui produse existent stocul lui va fi actualizat in baza de date.  POST-1.1: In cazul unui produs adaugat nou el va fi adaugat in lista de produse cu stocul aferent | | |
| **Normal flow** | **1.0 Adaugare stoc produs existent**   1. Managerul alege din meniu butonul de adaugare stoc. 2. Un submeniu se deschide unde managerul alege butonul de “Adaugare stoc produs existent”. 3. Se deschide o noua fereastra unde managerul poate alege la care produs sa adauge stoc si cantitatea adaugata. 4. Managerul apasa pe confirmare. 5. Apare un mesaj de succes    1. **Adaugare stoc produs nou** 6. Managerul alege din meniu butonul de adaugare stoc. 7. Un submeniu se deschide unde managerul alege butonul de “Adaugare produs nou”. 8. Se deschide o noua fereastra unde managerul toate informatiile despre noul produs si cantitatea adaugata. 9. Managerul apasa pe confirmare. 10. Apare un mesaj de succes | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **EC-1: Caz special: Daca se incearca confirmarea unei comenzi in timp ce ea este anulata de catre un agent de vanzari.Apare un mesaj de eroare.** | | |

Descriptions of template fields:

* **ID and name:** Title should be descriptive and should usually begin with a verb, e.g. order, calculate, input, etc. ID can have any format but must be unique among all use cases.
* **Primary actor:** Person that wishes to accomplish a goal through the use of the system. Only a single primary actor per use case.
* **Secondary actors:** Actors that have an interest in the completion of the goal but that do not directly interact with the system.
* **Description:** Concise description of the purpose of the use case.
* **Trigger:** Condition internal or external to the system that prompts the use case to start.
* **Preconditions:** Conditions that must be true before the use case starts. Each should be labeled with an ID unique to the use case.
* **Postconditions:** Conditions that must be true after the use case ends normally. Each should be labeled with an ID unique to the use case.
* **Normal flow:** Detailed step-by-step description of the logical flow of the use case. It should describe an explicit two way interaction, with the system prompting for input and the actor responding accordingly. Each step should be numbered.
* **Alternative flows:** Flows that achieve the same goal as the normal flow but are expected to be less common or lower priority.
* **Exceptions:** Conditions that result in the normal flow ending prematurely due to an unrecoverable condition in the system. The condition that causes the flow should be clearly stated, as should be any other decisions that the actor must make in this situation.

## Extra credit step: Traceability

For this extra step, you will add traceability information for each use case by adding a new field to the template:

|  |  |
| --- | --- |
| Method-level traces | <fully.qualified.ClassName>#<methodName>  ... |

Any method that implements the functionality described in the normal flow, alternative flow or exceptions should be included in this field. This means that the method that is initially executed and any methods of any classes that the work is delegated to should be included.

Examples for previous use cases:

UC-1:

|  |  |
| --- | --- |
| Method-level traces | my.company.ordering.MenuWidget#dateClicked  my.company.ordering.MenuWidget#completeOrder  my.company.ordering.InventoryInterface#checkInventory  ... |

UC-5:

|  |  |
| --- | --- |
| Method-level traces | my.company.payroll.PayrollInterface#checkEligibility  my.company.payroll.RegistrationForm#confirm  ... |